Real-Time Object Recognition in Color Imagery

Clark F. Olson
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Mail Stop 125-209, Pasadena, CA 91109

Abstract

Military test ranges containing unexploded ordnance due to live-fire testing and training exercises are a significant safety problem in many locations. Automated cleanup of such sites is a challenging goal. Towards this end, this paper describes algorithms to detect a type of ordnance in current usage using color imagery. The techniques are designed to run in real-time using off-the-shelf hardware for use on an unmanned ground vehicle. Our methodology is to quickly detect candidate locations using color and stereo information. Additional processing is then applied to the candidate locations in order to eliminate false alarms. This technique have been tested on a set of imagery from a live-fire test range with outstanding results.

1 Introduction

Cleanup of unexploded ordnance in military test ranges is a dangerous task that is currently performed by human technicians walking through the range, visually detecting unexploded ordnance, and performing remediation. Automating this task will both reduce the cost of cleaning the test ranges and eliminate the danger to the technicians. We are working towards this goal through the use of computer vision to recognize the ordnance from an unmanned ground vehicle.

Our methods have been designed towards a scenario where the vehicle traverses a test range at approximately 5 mph. As the vehicle traverses, the ordnance recognition system examines the terrain in front of the vehicle looking for instances of ordnance. In this work, we have concentrated on a type of ordnance called BLU-97, which is in current usage in U.S. military test ranges. The body of this type of ordnance is cylindrical and it is 20 centimeters long, with a 6 centimeter diameter. When new, the ordnance is bright yellow in color, but it is often weathered in practice on the test range.

We have designed algorithms for use in this scenario that can run in real-time (2-3 frames per second) using off-the-shelf hardware to ensure that the resulting system is timely and cost effective. Our methodology is to first detect candidate locations very quickly using color and stereo data. Once the candidates have been detected, we can apply additional computation to reduce false alarms, since much of the image has been eliminated from consideration. Finally, evidential reasoning is used to combine the information for the hypothesis detection and verification modules in order to make a final decision on each candidate.

These algorithms have been evaluated using a test set collected at a live-fire test range near Nellis Air Force Base. The results indicate that the system is capable of a very high rate of detection and few false positives, while requiring limited computation resources. We conclude with some lessons learned from this work.

2 Hypothesis detection

Our approach to detecting candidate locations of ordnance in the image is to first classify each pixel as ordnance-like or non-ordnance-like according to the color of the pixel. The candidates are then identified by locating blocks of connected ordnance-like pixels in the image. See Figure 1.

It has been shown that, under some simple assumptions about the scene, normalized color-space coordinates are independent of the scene geometry [2]:

$$(r, b, g) = \frac{(R, G, B)}{\sqrt{R^2 + G^2 + B^2}} \tag{1}$$

Thus, for scene points that are illuminated by the same spectral power distribution, the normalized image color is (largely) invariant to the orientation of the scene point, the orientation of the illumination, and the overall brightness of the illumination. If we can approximate sunlight as having a constant color and disregard specularity and inter-reflection effects, the normalized ordnance color is roughly constant. Of course, discoloration due to weathering and other effects, such as violations of the assumptions, will cause variation in the color. In our application, discoloration of the ordnance results in greater variation in the image chromaticities than illumination effects. We have thus chosen not to use a complex illumination compensation method.

The technique that we use to classify pixels is to define a polyhedron in the normalized color space, corresponding to the colors that we define to be ordnance-like. The polyhedron we use is a rectangular solid de-

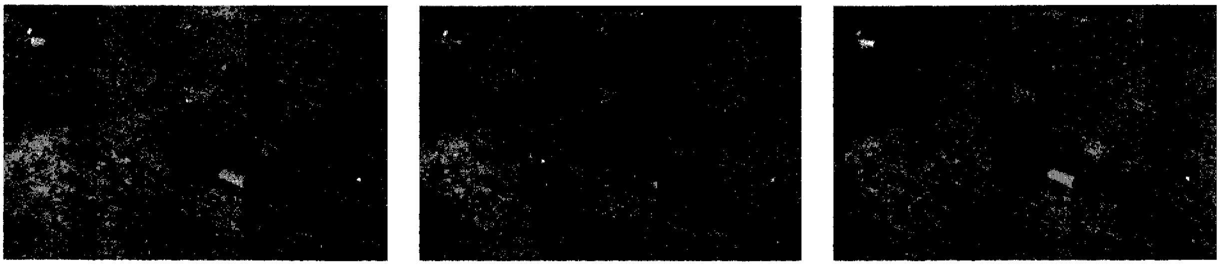
(a) (b) (c)

Figure 1: Hypothesis detection is performed by first classifying each pixel in the image as ordnance-like or non-ordnance-like according to its color. Hypothesis are then detected by finding large connected components using a threshold that varies with stereo range data. (a) Original image. (b) Classification results. The green pixels were classified as ordnance-like. (c) Candidates located.

fined by bounds on the normalized color coordinates:

$$t_r \geq r \geq T_r \tag{2}$$

$$t_g \geq g \geq T_g \tag{3}$$

$$t_b \geq b \geq T_b \tag{4}$$

This corresponds to a cone in the unnormalized color space with a six-sided, polygonal cross-section.

Now, we must determine the upper and lower bounds on these coordinates that will yield good classification of the ordnance-like pixels. We perform this through training using examples, using a method that approximates gradient descent search. At each iteration of the training, the number of errors that are made by the application of each of the inequalities is counted. Each threshold is then updated in the direction that would yield less overall errors. Training of the parameters has been performed on a dataset from the same site as the test set, but collected several months previously.

In practice, we compile the pixel classification yielded by the above method into a look-up table. Of course, this is a trade-off of time versus space. We have found very fast and accurate results can be obtained through the use of six bits to discretize each color. This incurs a storage requirement of 32K.

After classifying each of the pixels according to the above criterion, we detect the connected regions in the image that have been classified as ordnance-like. Detecting the connected components can be performed in linear time in the size of the image using a version of the union-find data structure [1]. This method uses a two-pass algorithm to detect the large, connected regions of ordnance-like pixels in the image. If the size of the region is larger than some threshold T (which we compute as a function of the range to the location), then the location is considered to be a candidate for further verification.

3 Verification

After detecting candidate ordnance locations, a variety of verification modules can be applied to the candidates in order to reduce the likelihood of detecting a false positive instance in the imagery. We can apply much more computation in these verification modules than in the initial hypothesis generation step since we have greatly reduced the area of the image in which we are interested.

3.1 Hypothesis resampling

A first step that is useful prior to applying the verification modules is to compute the dominant orientation in the image at the location of the hypothesis. We then resample the image, using the range and orientation information to transform a small area of the image around the candidate location such that the resampled image is at a canonical scale and orientation.

We detect the dominant orientation in the image by applying a simple gradient operator and then histogramming the gradient orientations found (weighted by the gradient magnitude) at each pixel. The histogram bin with the largest score is taken to be the dominant orientation of the hypothesis.

For each pixel in the resampled image, we then compute the corresponding location in the original image according to:

$$x = \frac{s_d}{s_h}(x_i - x_h)\cos\theta + \frac{s_d}{s_h}(y_i - y_h)\sin\theta \tag{5}$$

and

$$y = -\frac{s_d}{s_h}(x_i - x_h)\sin\theta + \frac{s_d}{s_h}(y_i - y_h)\cos\theta, \tag{6}$$

where θ is the dominant orientation, s_d is the desired scale, s_h is the scale of the hypothesis according to the

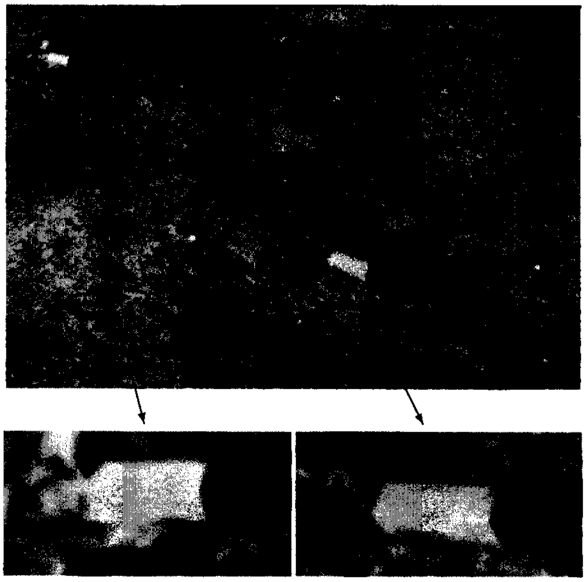

Figure 2: After candidates are located, they are resampled to a canonical size and orientation.

range data, x_i and y_i and the coordinates in the resampled image, and x_h and y_h are the center position of the hypothesis in the original image.

Figure 2 shows two examples of hypotheses that were resampled from an image.

3.2 Verification tests

Once the candidates have been resampled, we apply a series of tests to them, in order to determine which are actual instances of ordnance.

Gaussian filter. We apply a filter to the red band of the image consisting of the product of a Gaussian second derivative in the y-direction (across the cross-section of the ordnance) with a Gaussian in the x-direction (across the length of the ordnance). The filter is thus given by:

$$F(x,y) = \left(\frac{y^2 - \sigma_y^2}{2\pi\sigma_x\sigma_y^5}\right) e^{-\frac{x^2}{2\sigma_x^2} - \frac{y^2}{2\sigma_y^2}} \quad (7)$$

This yields high scores when the ordnance is present due to the yellow color of the ordnance in the center of the resampled candidates.

Parallel edge extraction. The gradients in the resampled candidate are histogrammed according to their orientation weighted by the gradient magnitude. If an orientation is found where the score is very high, this indicates either a single, very strong, straight edge is present, or a pair of strong parallel edges are present. Both of these cases are more likely if the candidate is an instance of the ordnance.

Height evaluation. Since we have range data, it is sometimes possible to detect the difference in height of the terrain at the location of an ordnance instance. A simple technique that we use is to examine the range data corresponding to the pixels in each candidate and determine the minimum and maximum heights present. The difference in these values measures the amount of height variation in the candidate window.

Contrast evaluation. When the candidate is an instance of the ordnance, we expect a significant gradient between the ordnance and the background. While false positives might also yield such a high gradient, this will not always be the case, and we can thus use this information to help discriminate between true positive and false positives.

Each of these tests yields a probabilistic score (as does the hypothesis generation stage) that feeds into an evidential reasoning process.

4 Evidential reasoning

Once the various verification modules have generated scores for each of the candidate ordnance locations, we must have some method for combining the scores into a single measure that can be used to evaluate each candidate. We use a linear opinion pool (see, for example, [3]), where the results of each measurement are combined according to some weighting factor that represents the confidence in the probability estimate that is generated.

Let H be the hypothesis that a certain candidate actually represents an ordnance instance. Each verification module v yields a probability value $P_v(H)$ that the hypothesis is correct and a weighting factor $W_v(H)$. We can combine the values from any two verification modules (for example v_1 and v_2) using the following relationships:

$$P_{v_1+v_2}(H) = \frac{W_{v_1}(H)P_{v_1}(H) + W_{v_2}(H)P_{v_2}(H)}{W_{v_1}(H) + W_{v_2}(H)} \quad (8)$$

$$W_{v_1+v_2}(H) = W_{v_1}(H) + W_{v_2}(H) \quad (9)$$

Since these equations are associative, it does not matter in which order the values are combined (the final result is the same). The candidate is finally accepted if, after combining all of the scores from the various verification modules, the probability value is above a pre-determined threshold.

5 Results

These techniques have been tested on a set of 350 images collected at a live-fire test range near Nellis Air

(a) (b) (c)

Figure 3: Results achieved on the Nellis Air Force Base data set. The boxes correspond to detected ordnance locations.

Force Base. The images were collected in a manner to simulate images from an unmanned ground vehicle. The data set thus consists of sequences of images captured at short intervals along nearly straight paths. Training for the hypothesis detection stage was performed on a set of images collected at the same site collected one year earlier. Overall, 324 instances of 75 different bombs appear in the test set. We have evaluated the techniques with respect to both the detection performance using a ROC curve and the computational requirements of the techniques through benchmarking on a workstation.

After the application of the recognition techniques to the complete data set, the performance was evaluated versus a manual identification of the instances present. Each bomb was detected in at least one of the images containing the bomb. Several false negatives occurred for instances that appeared at a significant distance from the camera and thus yielded small images of the bomb. In these cases, the bomb was always found when the camera traveled closer to it. In addition, 19 false positives are detected. Of the hypotheses reported, 92.6% are actually bombs.

Figure 3 shows results from this data set that include clutter and cases where the assumptions are violated by specularity, inter-reflection, and discolored ordnance. Despite the violation of various assumptions, the algorithm has little trouble discriminating between the ordnance and the background clutter.

In addition to the detection performance, we must consider the computation time required by the algorithms, since running in near real-time (2-3 frames per second) is necessary to accommodate a vehicle speed of of 5 MPH. The performance that we have observed on a workstation is 0.287 seconds per image to perform the stereo and hypothesis detection stages and 0.083 seconds per candidate to perform verification. A real-time system implementing these techniques with off-the-shelf hardware would likely run slightly slower.

6 Concluding remarks

Considerable work has been accomplished towards the generation of the real-time system to perform ordnance recognition. In the course of this work, several lessons have become apparent. First, it is crucial for the initial hypothesis detection techniques to be extremely fast in order to accommodate real-time performance of the system. None of the more complex techniques that we experimented with were even close to fast enough to support real-time operations. Next, stereo pre-processing yielded crucial information with respect to the scene depth. Without the depth information, the hypothesis threshold could not be set at a single value that yield low rates of both false positives and false negatives. Finally, shape was not an adequate discriminator by itself. We initially believed that shape would be a more robust discriminator than color due to possible discoloration of the ordnance. However, in many cases the shape was not reliably detected, resulting in many false positives. While the ordnance was sometimes discolored, it still remained recognizable with respect to the background.

Acknowledgements

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] M. B. Dillencourt, H. Samet, and M. Tamminen. A general approach to connected-component labeling for arbitrary image representations. *Journal of the ACM*, 39(2):253–280, April 1992.

[2] G. Healey. Segmenting images using normalized color. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(1):64–73, January/February 1992.

[3] F. Voorbraak. Combining evidence under partial ignorance. In *Proceedings of the First International Joint Conference on Qualitative and Quantitative Practical Reasoning*, pages 574–588, 1997.